

〔論説〕

確率的な交通障害発生における マルチエージェントの経路探索手法

愛知大学 岩田 員典
愛知工業大学 伊藤 暢浩
名古屋工業大学 社本 峻
名古屋工業大学 泉 泰介

第1章 はじめに

近年，世界各地で大規模な地震災害が頻発している．特に，2011年3月11日に発生した東北地方太平洋沖地震では，東北から関東にかけて甚大な被害を引き起こしたことは記憶に新しい．このような大規模な地震災害に対するマルチエージェント的アプローチとして，現在さまざまな取り組みが行なわれている．その取り組みの1つとして RoboCup Rescue Simulation プロジェクト^[1]が注目されている．RoboCup Rescue Simulation は，地震災害に伴う建物の火災および倒壊，道路の閉塞が発生した仮想的な被災都市において，災害救助隊（消防隊・救急隊・道路啓開隊）が災害救助活動を行なうシミュレーションである．RoboCup Rescue Simulation の中で，災害救助隊はエージェントとしてモデル化されており，マルチエージェントアプローチに基づいて災害救助活動のシミュレーションを行なうことが可能である．RoboCup Rescue Simulation にはさまざまな研究課題が含まれており，その中の1つにエージェントによる経路探索問題がある．RoboCup Rescue Simulation では，災害救助隊エージェントが災害救助活動を行なうために，災害救助活動の対象となる延焼中の建物や道路の瓦礫，

または市民の埋没した建物まで移動を行なう必要がある。このとき、災害救助活動をより迅速に行なうためには、エージェントの目的地までの経路をより効率的に探索する必要がある。

マルチエージェントシステムにおける経路探索問題の既存研究として、A* アルゴリズム^[2]やダイクストラ法^[3]などが知られている。これらのアルゴリズムはグラフネットワーク上の2つのノード間の最短経路を探索するアルゴリズムである。また、A* アルゴリズムを基に改良を施した D* アルゴリズム^[4]や Adaptive A* アルゴリズム^[5]などがある。これらの経路探索アルゴリズムは、計算時間の高速化に重点を置いている。しかし、災害救助活動や避難のための経路探索では、道路が全て使える状況は少なく、道路に障害が発生する可能性がある。すなわち、環境（道路情報）に関する事前知識が正確でなくなっていることを想定しなければならない。一方で、平常時での周辺環境調査等により災害時における環境変化についてある程度の予想が可能であると仮定することは妥当であると考えられる。よって、災害時における移動計画においては、これらの不確実性と予想可能性を考慮して経路選択をすることが望ましい。また、計算時間そのものよりも、目的地へ到達するまでの時間を短くすることが重要となる。そのため、移動時間に比例する総移動コストも重要となる。

そこで本論文では、確率的障害が発生することを想定し、より効率よく目的地まで移動するために、道路にどれくらいの確率で障害が発生するかを示す「障害発生率」と、最短経路に障害があったときの代替経路である「迂回経路」を利用することを考える。具体的には、障害による経路の封鎖確率、および迂回経路への切り替え容易性を考慮した評価関数に基づく新しい A* アルゴリズムを提案し、その性能評価を行う。

第2章 諸定義

本章では、経路探索問題に共通して用いられるデータ構造などを定義する。

経路探索がおこなわれる重み付きグラフ G を以下のように定義する.

定義 1 グラフ

$$G=(V, E, C)$$

ここで V はノード (node) の集合, E はノードとノードをつなぐ辺 (edge) である.

また, C は辺にコストを与える関数である. それぞれ, 以下のように定義する.

定義 2 ノードの集合

$$V = \{v_1, v_2, \dots, v_n\}$$

定義 3 辺の集合

$$E = \{e_{i,j} \mid v_i, v_j \in V, v_i \text{ と } v_j \text{ との間に辺が存在する} \}$$

定義 4 辺のコスト関数

$$C: E \rightarrow R^+$$

ここで, 辺 $e_{i,j}$ に与えられたコストは $c_{i,j}$ とする.

グラフ G におけるノード v_s から v_g までの経路を以下のように定義する.

定義 5 経路

経路はノード列 $(v_s, v_{s+1}, \dots, v_{g-1}, v_g)$ で表す. 尚, 各 v_i から v_{i+1} には, 辺 $e_{i,i+1}$ が存在しなければならない.

また, 経路 $(v_s, v_{s+1}, \dots, v_{g-1}, v_g)$ のコストを以下のように定義する.

定義 6 経路のコスト

$$c_{s,s+1} + c_{s+1,s+2} + \dots + c_{g-2,g-1} + c_{g-1,g}$$

ノード v_s からノード v_g までの最短経路とそのときのコストである最短コス

トを以下のように定義する.

定義 7 最短経路

ノード v_s からノード v_g までの全ての経路の中で, 経路のコストが最小のもの.

定義 8 最短コスト

最短経路のコストを最短コストと呼ぶ.

探索を開始するノード (開始ノード) から目的地となるノード (目的地ノード) へ移動する対象をエージェントと呼び, 各エージェントの能力および動作を以下のように定義する.

定義 9 エージェントの能力と動作

- エージェントは, 与えられたアルゴリズムにしたがって, 移動先ノードを決定する,
- エージェントが同一環境に複数体存在するとき, 全エージェントは同時に動作する,
- エージェント同士は自由かつ同時に通信可能である,
- エージェントは同一グラフ上に複数存在することがある.

辺, ノードにおける障害とその障害に関する情報を以下のように定義する.

定義10 障害

辺 (もしくはノード) に何らかの問題が発生し, エージェントが通過できなくなる状態を, 辺 (もしくはノード) の障害とする.

定義11 障害情報

各辺（もしくは各ノード）の障害の有無に関する情報を障害情報とする。

障害情報を得たエージェントは目的地に到達するために迂回する。その迂回経路を以下のように定義する。

定義12 迂回経路

最短経路上のある辺が障害により通行不可能であると仮定したときの（次善の）最短経路を迂回経路とする。

最後に、最短経路探索問題について定義する。

定義13 最短経路問題

エージェントが探索を開始するノード（開始ノード）から目的地となるノード（目的地ノード）までの最短経路を探索する問題尚、エージェントが複数存在する場合は、「全エージェントが開始ノードから目的地ノードまで移動完了するのにかかる時間」を最小にする経路を探索する問題とする。

第3章 最短経路探索問題の既存研究

3.1 A* アルゴリズム

A* アルゴリズム^[2]は、1968年に Hart らによって開発されたアルゴリズムであり、人工知能における経路探索問題の代表的なアルゴリズムとなっている。A* アルゴリズムでは、開始ノードから目的地ノードへ至る最短コストの推定値に基づいて経路探索を進める。これにより、経路の全探索をせずに短い計算時間で目的地ノードまでの最適経路を探索することができる。

3.1.1 A* アルゴリズムの概要

A* アルゴリズムは、あるノードから隣接するノードを選択する時に、候補となる隣接ノードから目的地ノードまでの最短コストの推定値が最も小さくなるノードを選択して経路探索を進めていく方法である。ここで、最短コストの推定値について説明する。探索経路上にあるノードを p とする。開始ノードからノード p を通り、目的地ノードまで至る経路の最短コスト $f(p)$ を、以下のように表す。

$$f(p) = g(p) + h(p)$$

ここで、 $g(p)$ と $h(p)$ は以下の意味を持つ。

- $g(p)$: 開始ノードからノード p までの最短コスト
- $h(p)$: ノード p から目的地ノードまでの最短コスト

探索がノード p まで進んだ場合を考えると、開始ノードからノード p までの最短コスト $g(p)$ は求められているが、 $h(p)$ は探索を進めていかなければ求めることができない。そこで、 $h(p)$ には推定値として $h^*(p)$ を用いて、開始ノードからノード p を通り目的地ノードへ至る最短コストの推定値を以下のように表す。

$$f^*(p) = g(p) + h^*(p)$$

このとき $h^*(p)$ のことをヒューリスティック関数とよぶ。そして、このアルゴリズムは $h^*(p)$ が条件 $\forall p, 0 \leq h^*(p) \leq h(p)$ を満たすとき、求まる経路が開始ノードから目的地ノードまでの最短経路であることが保証されている。これを A* アルゴリズムと呼ぶ。したがって、現在経路探索が進んでいるノード

から次の経路となるノードを選択する時に、候補となるノードについて $f^*(p)$ の値を求め、 $f^*(p)$ の値が最小になるノードを経路として選択することで経路探索を進める。代表的な $h^*(p)$ の値としては、目的地までのユークリッド距離などがある。

3.2 A* アルゴリズムの派生

本節では、現在までに考案されてきた A* アルゴリズムの改良について簡単に説明する。

3.2.1 D* アルゴリズム

D* アルゴリズム^[4]は、1994年に Anthony Stentz によって提案された経路探索アルゴリズムである。D* アルゴリズムでは、目的地ノードから A* アルゴリズムを実行し、各ノードで目的地方向の辺を記録しておくことで、現在地から目的地へ移動中になんらかの障害が起きた場合に、A* アルゴリズムによる探索を繰り返すよりも短い計算時間で迂回経路を求めることができる。

3.2.2 Adaptive A* アルゴリズム

Adaptive A* アルゴリズム^[5]は、2006年に Sven らによって提案された経路探索アルゴリズムである。Adaptive A* アルゴリズムでは、通行可能な辺の数が減少する場合に、A* アルゴリズムによる探索を繰り返すよりも短い計算時間で迂回経路を求めることができる。

3.3 既存アルゴリズムに対する検討

本章で述べたように、現在の A* アルゴリズムの改良は計算時間に重きをおいているものがほとんどである。しかし、計算時間そのものが高速であっても、通行不能な障害へ到達してしまうと、迂回したり、戻ったりする分だけ、目的地までの総移動コストが大きくなり、目的地へ到達するのが遅くなってしまう。

そこで、本論文では経路障害に到達することを確率的に避け、最終的に目的地に到達するまでにかかる時間を短縮するアルゴリズムを、次章で提案する。

第4章 提案手法

本章では、本論文が提案する経路探索手法について説明する。まずは本論文が対象とする問題設定について説明し、次に本論文の提案手法について説明する。

4.1 問題設定

本節では、本論文が対象とする問題の環境とエージェントについて説明する。

4.1.1 既存研究との相違点

既存研究では、入力グラフに移動障害が発生することを考慮していない場合が多い。また、入力グラフに移動障害が発生することを考慮している研究であっても、障害情報が簡単に入手できる設定となっている。しかし、地震災害等で道路に障害が発生した場合、障害を考慮していない経路探索は効率的でないことが多く、障害情報が簡単に入手できない場合も多い。そこで、本論文では、入力グラフの辺に障害が発生し、障害情報が簡単に入手できないような問題設定をおこなう。また、エージェントは現実世界の人間を想定しており、十分に広い道路を移動することを想定しているので、エージェント同士の衝突や渋滞は考慮しないものとする。

4.1.2 環境

本論文では、経路探索をおこなう障害発生率を持つ重み付きグラフ G' を以下のように定義する。

定義14 障害発生率を持つグラフ

$$G'=(V, E, C, B)$$

V, E, C はそれぞれ定義2, 3, 4と同じものである。また、辺に障害発生率を与える関数 B を以下のように定義する。

定義15 障害発生率

$$B: E \rightarrow R^+$$

ここで、辺 $e_{i,j}$ 与えられた障害発生率は $b_{i,j}$ とする。

本論文で扱う経路障害はシミュレーション開始時に、「障害発生率」に基づいて各辺に発生する。これ以降、経路障害が発生した辺を障害辺と呼ぶ。また、経路障害はシミュレーション開始時点で決定され、シミュレーション中に増減することはない。

4.1.3 エージェント

本論文では、エージェントの能力および動作を以下のように定める。

- 与えられたアルゴリズムにしたがって、移動先ノードを決定する。
- 同一環境に複数体エージェントが存在するとき、全エージェントは同時に動作する。
- エージェント同士は同時に通信可能であり、受信可能メッセージ数やメッセージ長の制限はない。
- 初期状態で障害情報を含まないグラフに関する情報を取得できる。
- 障害辺に接続しているノードに到達するまで、経路障害を発見できない。
- 複数のエージェントが、同時に同じノード（または辺）に移動しても衝突・渋滞は発生しない。

4.2 アルゴリズム

本論文では、A* アルゴリズムを基にアルゴリズムを設計する。3章で説明したように、目的地までの到達時間を短くするためには、計算時間と目的地までの総移動コストを小さくする必要がある。しかし、移動中に経路障害へ到達してしまうと、目的地への総移動コストが大きくなってしまう。また、経路障害は、障害辺に接続しているノードに到達しないと発見できない。そこで、本論文では障害発生率と迂回経路を利用することによって、A* アルゴリズムを改良し、この問題の解決を目指す。

4.2.1 A* アルゴリズムの改良

本論文では、A* アルゴリズムで用いられる $h^*(p)$ の計算方法を提案する。 $h^*(p)$ の計算方法を以下に示す。ここでは、ノード p から目的地までの最短経路を (v_1, v_2, v_3, \dots) とする。

1. ノード p から目的地までの最短コストを計算する。
2. 最短コストで目的地へ到達できる確率を計算する。
3. 1と2の積を計算する。
4. 各ノード v_i において v_i から v_{i+1} が通れないと仮定した場合の最短コスト（迂回経路におけるコスト：迂回経路長とする）を計算する。
5. 3を基に次の計算を行う。

$$\begin{aligned} & b_{1,2} \times \text{ノード1の迂回経路長} \\ & + (1 - b_{1,2}) \times b_{2,3} \times \text{ノード2の迂回経路長} \\ & + (1 - b_{1,2}) \times (1 - b_{2,3}) \times b_{3,4} \times \text{ノード3の迂回経路長} \\ & + \dots \end{aligned}$$

6. 3と5の和を $h^*(p)$ とする。

4.2.2 動作例

図4.1に示すグラフにおいて、ノード7からノード8までの最短経路に関して提案手法を用いて探索する例を考える。尚、簡単のため一部の障害発生率のみ示す。

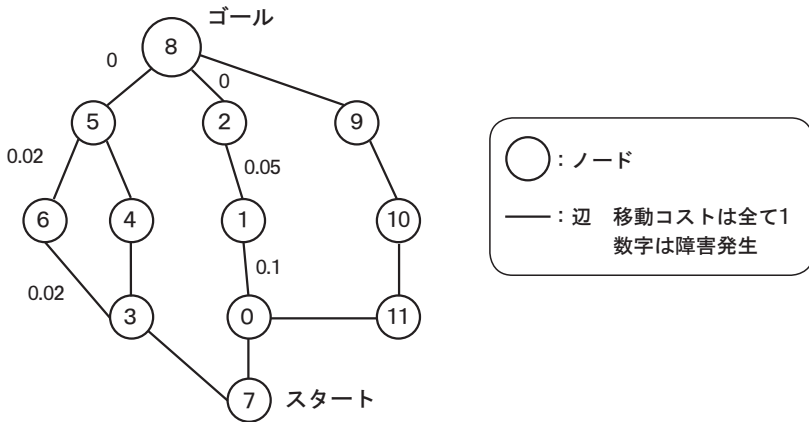


図 4.1: 入力グラフ

アルゴリズムは以下の流れに沿って行われる。

- 現在のノードに隣接しているノード3について $h^*(3)$ を計算する。
 - 最短コストで目的地へ到達できる確率 \times 最短コスト

$$0.98 \times 0.98 \times 1 \times 3 = 2.8812$$
 - (迂回経路を選択しなければならない確率 \times 迂回経路のコスト) の合計

$$0.02 \times 3 + 0.98 \times 0.02 \times 4 + 0.98 \times 0.98 \times 0 \times 7 = 0.1384$$
 - $h^*(3) = 2.8812 + 0.1384 = 3.0196$
- 同様に、ノード0について $h^*(0)$ を計算する。ここでは3.19となる。
- 各ノードの $f^*(p)$ を計算する。
 - $f^*(3) = g(3) + h^*(3) = 1 + 3.0196 = 4.0196$

$$(b) f^*(0) = g(0) + h^*(0) = 1 + 3.19 = 4.194$$

4. 現在のノードに隣接しているノードの中で、 $f^*(p)$ の値が一番小さいノードへ移動する。ここではノード3へ移動する。
5. 現在のノードに隣接しているノード6とノード4とノード7について $f^*(p)$ を計算する。
6. ノード6の $f^*(p)$ とノード4の $f^*(p)$ の値が同じになるが、ここではノード6へ移動する。
7. 現在のノードに隣接しているノード5とノード3について $f^*(p)$ を計算する。
8. ここではノード5へ移動する。
9. 現在のノードに隣接しているノード8は目的地なので、ノード8へ移動する。
10. 目的地へ到達したので、アルゴリズム終了となる。

4.3 A* アルゴリズムと提案手法の複合について

本論文では、エージェントが複数の場合も実験をおこなう。その場合、A* アルゴリズムと提案手法のほかに、A* アルゴリズムと提案手法を組み合わせた手法が効果的な可能性が有る。そこで、本論文ではA* アルゴリズムと提案手法を組み合わせた手法でも実験をおこなう。この場合、エージェントの半数がA* アルゴリズムで動作し、残りの半数が提案手法で動作するものとする。また、本論文ではA* アルゴリズムと提案手法を組み合わせた手法を複合と呼ぶ。

4.4 理論値

本論文では、提案手法がどの程度、総移動コストの最小値に近づけたかを判断するため、最初から全ての障害情報を持っている状態で計算した最短経路長も計算する。また、最初から全ての障害情報を持っている状態で計算した最短経路長を理論値と呼ぶ。

第5章 提案手法の評価

5.1 実験目的

本論文で提案した経路探索手法が、既存手法の A* アルゴリズムや理論値と比べて、どの程度の総移動コストになるか、また、どの程度の計算時間になるかを確認するために実験をおこなう。

5.2 本実験であつかう A* アルゴリズムについて

本実験であつかう A* アルゴリズムの推定値 $h^*(p)$ には、実際のノード p からゴールまでの移動コストを用いた。つまり、ダイクストラ法と同じ方法で計算するものとする。

5.3 実験環境

実験には1台の計算機を使用した。その計算機のスペックは表5.1の通りである。また、プログラム言語には C++ を使用し、コンパイル環境には Visual Studio 2010を使用した。

表5.1: 実験に用いた計算機のスペック

OS	Windows 7
CPU	AMD Athlon X2 Dual Core Processor 5200+
Memory	3GB

5.4 有意差の検定

総移動コストの平均・標準偏差だけでは、提案手法が優れているかどうか判断しづらい。そこで、本論文では、実験結果の解析のために、有意差の検定をおこなった。ここでは、実験結果のヒストグラムより、実験結果は明らかに正規分布にしたがっておらず、3標本以上の検定となるので、ノンパラメトリック検定のスティール・ドゥワース法^[8]を用いた。また、帰無仮説を「平均値に差がない」とし、棄却率は1%とした。

5.5 確認実験

5.5.1 確認実験内容

まず、提案手法が A* アルゴリズムに比べて、総移動コストが小さくなるかどうかを簡単な入力グラフで確認する。確認実験に使用した入力グラフを図 5.1, 図 5.2 に示す。また、各確認実験の詳細を表 5.2 に示す。さらに、エージェントが複数の場合、全エージェントで開始ノードと目的地ノードは同一だが、スタートするタイミングが違うものとする。

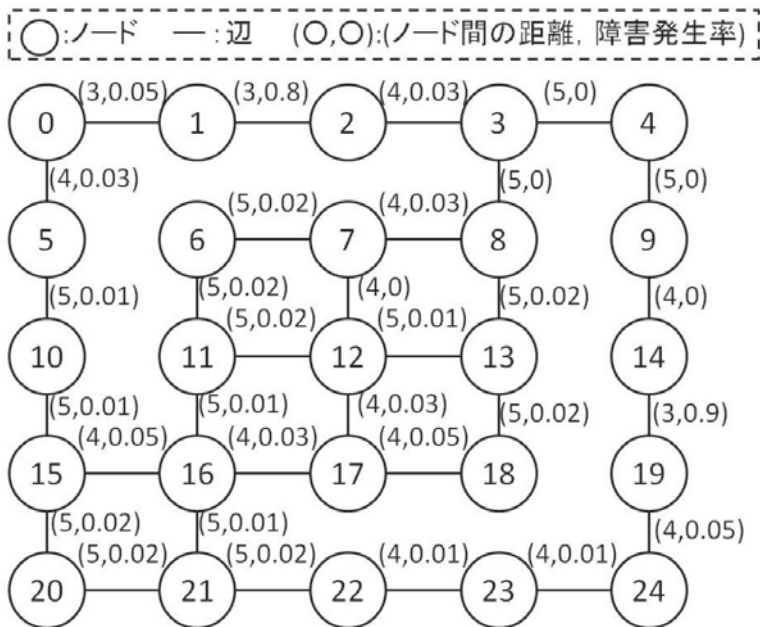


図 5.1 グラフ1

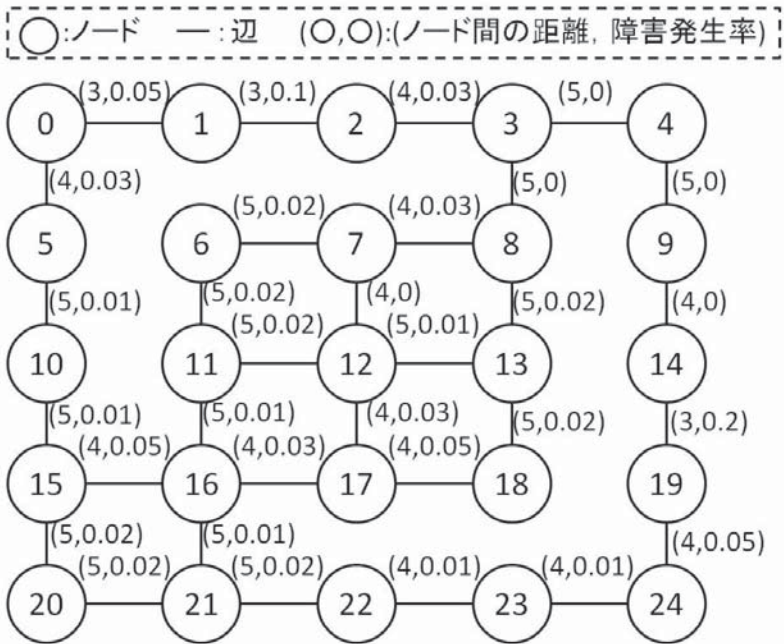


図 5.2 グラフ2

表5.2: 各確認実験の設定

実験番号	グラフ	開始ノード	目的地ノード	エージェント数	実験回数
1-1	グラフ1	ノード20	ノード4	1	1000
1-2	グラフ2	ノード20	ノード4	10	1000
2-1	グラフ1	ノード20	ノード4	1	1000
2-2	グラフ2	ノード20	ノード4	10	1000

5.5.2 確認実験結果

5.5.2.1 実験 1-1

エージェントの総移動コストの平均・標準偏差と、実験に要した平均計算時間を表5.3に示す.

表5.3: 実験1-1におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均	標準偏差	平均計算時間 (s)
A*	66.499	12.23	0.00021
提案手法	35.958	2.192	0.00030
理論値	34.927	-	-

実験1-1におけるエージェントの総移動コストのヒストグラムを図5.3に示す.

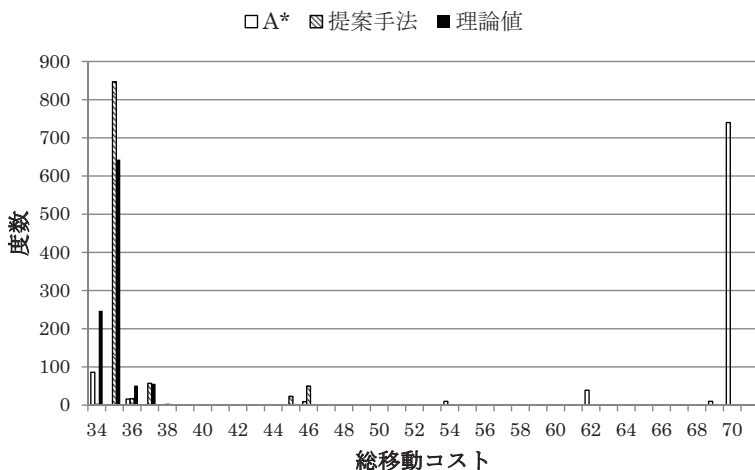


図 5.3: 実験 1-1の総移動コストのヒストグラム

A* アルゴリズムと提案手法に関してコストの平均値に対しての検定の結果, $t = 34.106$, $P = 0.000$ となり帰無仮説は棄却された. よって, A* アルゴリズムと提案手法におけるコストの平均値には有意差が有る.

5.5.2.2 実験 1-2

エージェントの総移動コストの平均・標準偏差と, 実験に要した平均計算時間を表5.4に示す.

表5.4: 実験1-2におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均	標準偏差	平均計算時間 (s)
A*	42.327	3.716	0.00112
提案手法	35.393	1.050	0.00186
複合	40.516	2.790	0.00156
理論値	34.934	-	-

実験1-2におけるエージェントの総移動コストのヒストグラムを図5.5に示す。

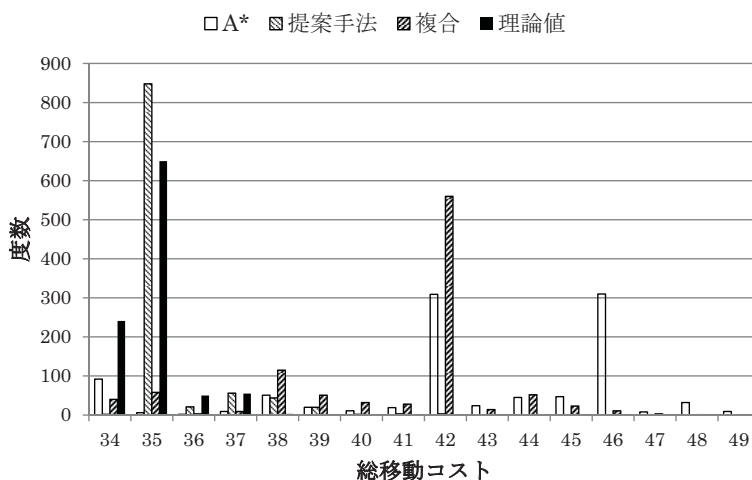


図 5.4: 実験 1-2の総移動コストのヒストグラム

また、3つの手法に関してコストの平均値に対する有意差検定の結果を表5.5に示す。

表5.5: 実験1-2におけるコストの平均の有意差検定結果

比較対象	t	P	有意差
A*：提案手法	3.1954	0.000	有
A*：複合	11.505	0.000	有
提案手法：複合	31.635	0.000	有

5.5.2.3 実験 2-1

エージェントの総移動コストの平均・標準偏差と、実験に要した平均計算時間を表5.6に示す。

表5.6: 実験2-1におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均	標準偏差	平均計算時間 (s)
A*	42.370	14.671	0.00031
提案手法	35.945	5.399	0.00017
理論値	34.090	-	-

実験2-1におけるエージェントの総移動コストのヒストグラムを図5.5に示す。

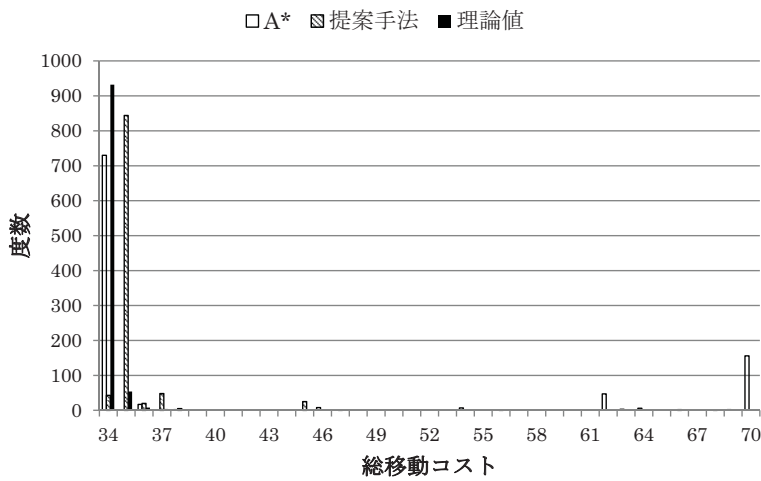


図 5.5: 実験 2-1の総移動コストのヒストグラム

A* アルゴリズムと提案手法に関してコストの平均値に対しての検定の結果, $t = 18.175$, $P = 0.000$ となり帰無仮説は棄却された. よって, A* アルゴリズムと提案手法におけるコストの平均値には有意差が有る.

5.5.2.4 実験 2-2

エージェントの総移動コストの平均・標準偏差と、実験に要した平均計算時間を表5.7に示す。

表5.7: 実験2-2におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均	標準偏差	平均計算時間 (s)
A*	35.432	2.736	0.00096
提案手法	35.237	1.024	0.00163
複合	35.178	2.018	0.00154
理論値	34.071	-	-

実験2-2におけるエージェントの総移動コストのヒストグラムを図5.6に示す。

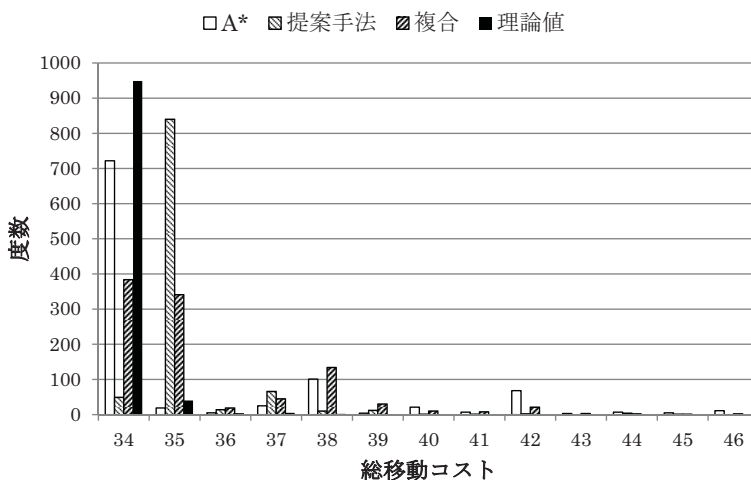


図 5.6: 実験2-2の総移動コストのヒストグラム

また、3つの手法に関してコストの平均値に対する有意差検定の結果を表5.8に示す。

表5.8: 実験2-2におけるコストの平均の有意差検定結果

比較対象	t	P	有意差
A*: 提案手法	19.004	0.000	有
A*: 複合	9.977	0.000	有
提案手法: 複合	16.974	0.000	有

5.5.3 考察

確認実験の結果より、提案手法は既存手法よりも平均移動コストは小さくなることを示している。ただし、その経路の計算には提案手法の方が時間を要する。しかし、その差は移動コストに比べて小さいため影響は無視できる。つまり、このような環境において、提案手法が効果的である。そこで、次節から実際の地図データから生成したグラフでも、効果があるかどうかを確認するために実験をおこなう。

5.6 応用実験

5.6.1 応用実験内容

応用例として、実際の地図データから生成したグラフでも提案手法の効果があることを確認する。

本論文では、国土地理院の発行している2003年度版地図データ^[6]から入力グラフ $G' = (V, E, C, B)$ を生成した。ここでは、 V は交差点の集合、 E は交差点を繋ぐ道路の集合、 C は交差点と交差点の距離の集合、 B は道路の障害発生率とする。また、道路の障害発生率 B は名古屋市が発行している「あなたの街の地震マップ」^[7]の液状化危険度を基に設定した。各応用実験の詳細を表5.9に示す。また、エージェントが複数の場合、全エージェントで開始ノードと目的地ノードは同一だが、スタートするタイミングが違うものとする。

表5.9: 各応用実験の設定

実験番号	地区	開始位置	目的地	エージェント数	実験回数
3	昭和区	昭和区役所	鶴舞公園	1	1000
4	中村区	中村区役所	中村公園	1	1000
5-1	中村区	松原町3丁目	中村公園	1	1000
5-2	中村区	松原町3丁目	中村公園	10	1000
5-3	中村区	松原町3丁目	中村公園	100	1000
5-4	中村区	松原町3丁目	中村公園	20	1000
5-5	中村区	松原町3丁目	中村公園	30	1000
6-1	西区	南堀越1丁目	枇杷島公園	1	1000
6-2	西区	南堀越1丁目	枇杷島公園	10	1000
6-3	西区	南堀越1丁目	枇杷島公園	20	1000

5.6.2 応用実験結果

5.6.2.1 実験3

エージェントの移動コスト (m) の平均・標準偏差と、実験に要した平均計算時間を表5.10に示す.

表5.10: 実験3におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	998.50	122.8	0.00185
提案手法	998.35	122.6	0.00789
理論値	932.38	-	-

エージェントの総移動コスト (m) のヒストグラムを図5.7に示す.

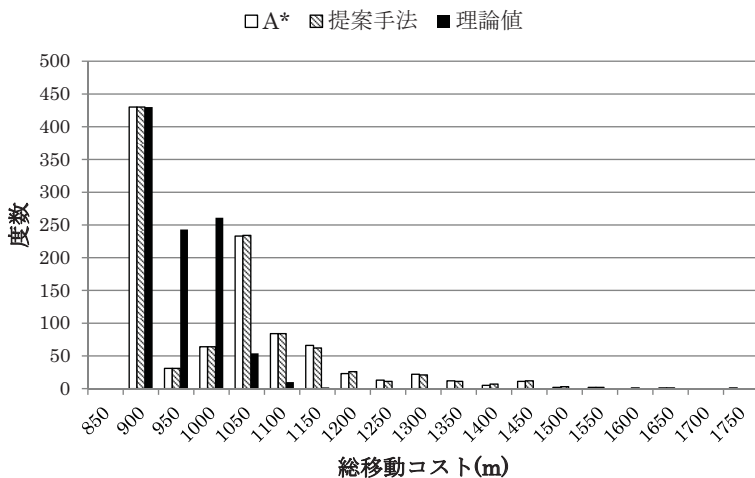


図5.7: 実験3の総移動コストのヒストグラム

A* アルゴリズムと提案手法に関してコストの平均値に対しての検定の結果, $t = 0.054$, $P = 0.998$ となり帰無仮説は棄却されなかった. よって, A* アルゴリズムと提案手法におけるコストの平均値には有意差が有るとは言えない.

5.6.2.2 実験4

エージェントの総移動コスト (m) の平均・標準偏差と, 実験に要した平均計算時間を表5.11に示す.

表5.11: 実験4におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	2390.45	363.9	0.00626
提案手法	2355.78	325.6	0.78707
理論値	1980.52	-	-

エージェントの移動コスト (m) のヒストグラムを図5.8に示す.

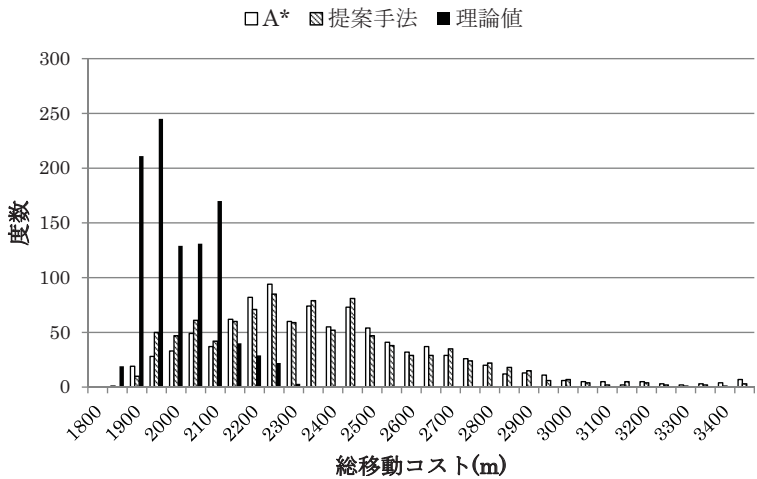


図 5.8: 実験4の総移動コストのヒストグラム

A* アルゴリズムと提案手法に関してコストの平均値に対しての検定の結果, $t = 1.807$, $P = 0.167$ となり帰無仮説は棄却されなかった. よって, A* アルゴリズムと提案手法におけるコストの平均値には有意差が有るとは言えない.

5.6.2.3 実験5-1

エージェントの移動コスト (m) の平均・標準偏差と, 実験に要した平均計算時間を表5.12に示す.

表5.12: 実験5-1におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	2209.74	411.41	0.00553
提案手法	1631.94	237.29	0.05872
理論値	1518.90	-	-

エージェントの総移動コスト (m) のヒストグラムを図5.9に示す.

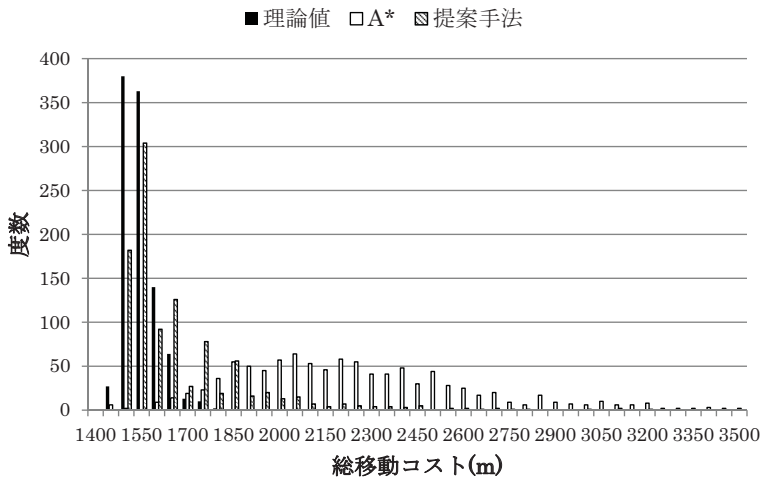


図5.9: 実験5-1の総移動コストのヒストグラム

A* アルゴリズムと提案手法に関してコストの平均値に対しての検定の結果, $t = 33.116$, $P = 0.000$ となり帰無仮説は棄却された. よって, A* アルゴリズムと提案手法におけるコストの平均値には有意差が有る.

5.6.2.4 実験 5-2

エージェントの移動コスト (m) の平均・標準偏差と, 実験に要した平均計算時間を表5.13に示す.

表5.13: 実験5-2におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	1697.09	123.61	0.02049
提案手法	1621.50	178.60	0.31606
複合	1633.02	99.90	0.10512
理論値	1517.91	-	-

エージェントの総移動コスト (m) のヒストグラムを図5.10に示す.

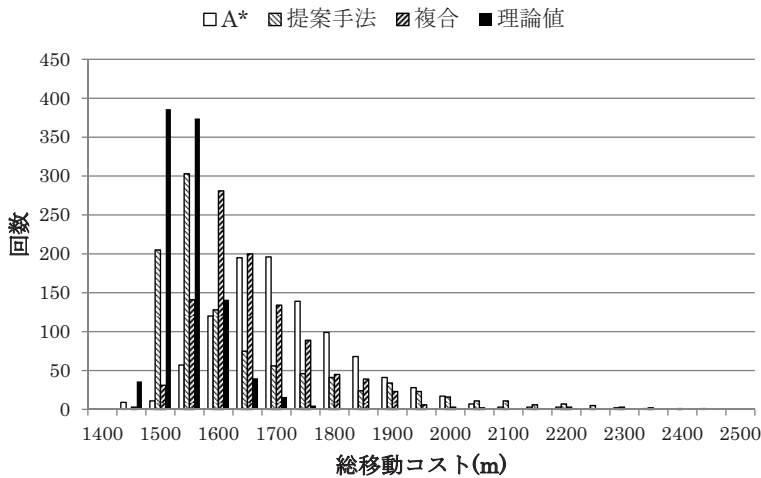


図5.10: 実験5-2の総移動コストのヒストグラム

また、3つの手法に関してコストの平均値に対する有意差検定の結果を表5.14に示す。

表5.14: 実験5-2におけるコストの平均の有意差検定結果

比較対象	<i>t</i>	<i>P</i>	有意差
A*：提案手法	17.239	0.000	有
A*：複合	13.346	0.000	有
提案手法：複合	10.340	0.000	有

5.6.2.5 実験5-3

エージェントの移動コスト (m) の平均・標準偏差と、実験に要した平均計算時間を表5.15に示す。

表5.15: 実験5-3におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	1537.18	56.28	0.17054
提案手法	1632.04	255.3	3.14919
複合	1529.91	55.07	0.67886
理論値	1519.17	-	-

エージェントの総移動コスト (m) のヒストグラムを図5.11に示す.

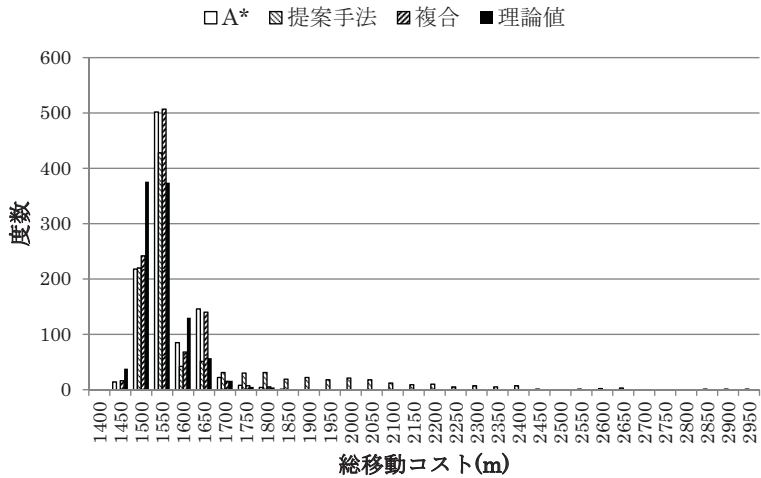


図5.11: 実験5-3の総移動コストのヒストグラム

また, 3つ手法に関してコストの平均値に対する有意差検定の結果を表5.16に示す.

表5.16: 実験5-3におけるコストの平均の有意差検定結果

比較対象	t	P	有意差
A* : 提案手法	2.296	0.000	有とは言えない
A* : 複合	5.086	0.000	有
提案手法 : 複合	5.592	0.000	有

5.6.2.6 実験5-4

エージェントの移動コスト (m) の平均・標準偏差と、実験に要した平均計算時間を表5.17に示す。

表5.17: 実験5-4におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	1062.39	91.04	0.03772
提案手法	1032.76	88.14	0.43829
複合	1049.27	86.83	0.16579
理論値	963.05	-	-

エージェントの総移動コスト (m) のヒストグラムを図5.12に示す。

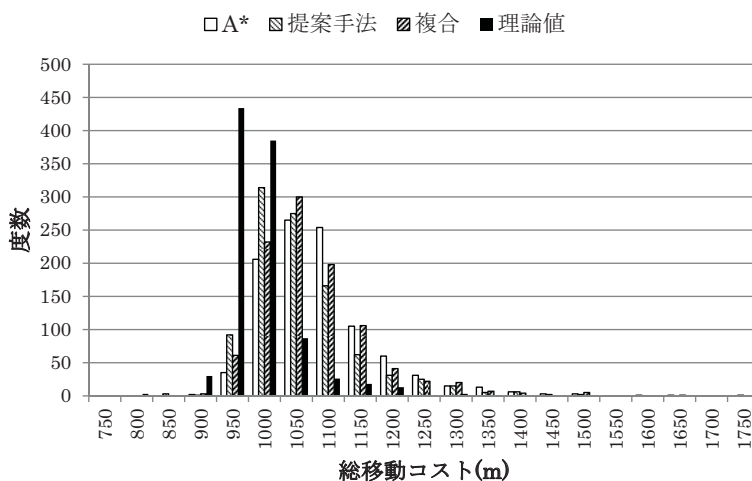


図5.11: 実験5-4の総移動コストのヒストグラム

また、3つの手法に関してコストの平均値に対する有意差検定の結果を表5.18に示す。

表5.18: 実験5-4におけるコストの平均の有意差検定結果

比較対象	t	P	有意差
A* : 提案手法	9.277	0.000	有
A* : 複合	3.744	0.000	有
提案手法 : 複合	5.705	0.000	有

5.6.2.7 実験5-5

エージェントの総移動コスト (m) の平均・標準偏差と、実験に要した平均計算時間を表5.19に示す。

表5.19: 実験5-5におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	1007.80	85.58	0.05782
提案手法	993.70	82.23	0.43575
複合	990.04	74.28	0.17549
理論値	930.66	-	-

エージェントの総移動コスト (m) のヒストグラムを図5.13に示す。

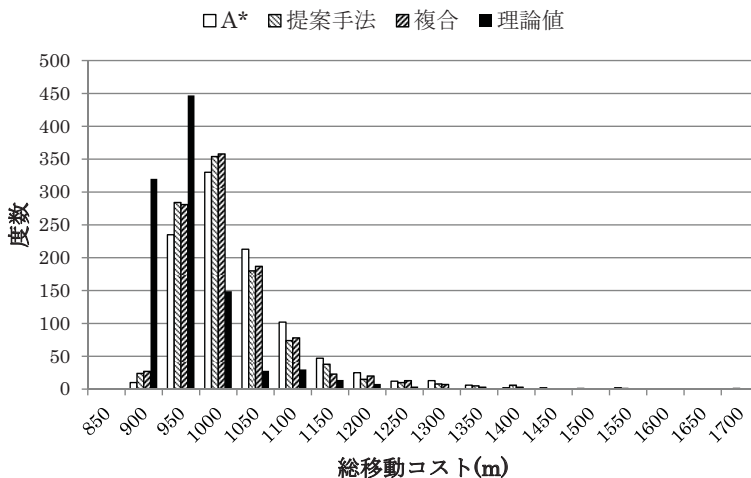


図5.13: 実験5-5の総移動コストのヒストグラム

また、3つの手法に関してコストの平均値に対する有意差検定の結果を表5.20に示す。

表5.20: 実験5-5におけるコストの平均の有意差検定結果

比較対象	t	P	有意差
A*：提案手法	4.677	0.000	有
A*：複合	5.290	0.000	有
提案手法：複合	0.599	0.932	有とは言えない

5.6.2.8 実験 6-1

エージェントの移動コスト (m) の平均・標準偏差と、実験に要した平均計算時間を表5.21に示す。

表5.21実験6-1におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	2210.89	485.2	0.00586
提案手法	1916.49	397.5	0.10926
理論値	1706.52	-	-

エージェントの総移動コスト (m) のヒストグラムを図5.14に示す。

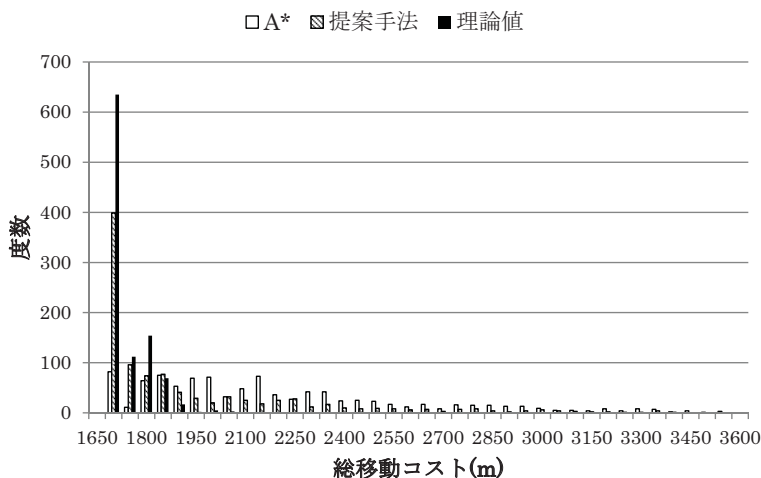


図5.14: 実験6-1の総移動コストのヒストグラム

A* アルゴリズムと提案手法に関してコストの平均値に対しての検定の結果, $t = 18.731, P = 0.000$ となり帰無仮説は棄却された. よって, A* アルゴリズムと提案手法におけるコストの平均値には有意差が有る.

5.6.2.9 実験6-2

エージェントの移動コスト (m) の平均・標準偏差と, 実験に要した平均計算時間を表5.22に示す.

表5.22: 実験6-2におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	1848.07	148.1	0.02639
提案手法	1824.26	227.9	0.41419
複合	1798.80	126.0	0.15669
理論値	1704.31	-	-

エージェントの総移動コスト (m) のヒストグラムを図 5.15に示す.

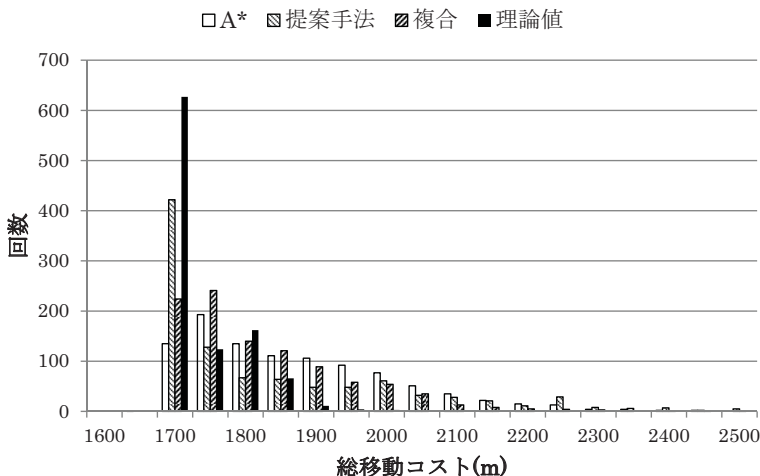


図5.15: 実験6-2の総移動コストのヒストグラム

また, A* アルゴリズムと提案手法に関してコストの平均値に対する有意差検定の結果を表5.23に示す.

表5.23: 実験6-2におけるコストの平均の有意差検定結果

比較対象	t	P	有意差
A* : 提案手法	9.671	0.000	有
A* : 複合	8.168	0.000	有
提案手法 : 複合	4.535	0.000	有

5.6.2.10 実験6-3

エージェントの移動コスト (m) の平均・標準偏差と、実験に要した平均計算時間を表5.24に示す。

表5.24: 実験6-3におけるコストの平均・標準偏差と平均計算時間

アルゴリズム	平均 (m)	標準偏差 (m)	平均計算時間 (s)
A*	1773.49	95.77	0.04951
提案手法	1768.35	146.7	0.63650
複合	1749.46	87.65	0.21919
理論値	1704.07	-	-

エージェントの総移動コスト (m) のヒストグラムを図5.16に示す。

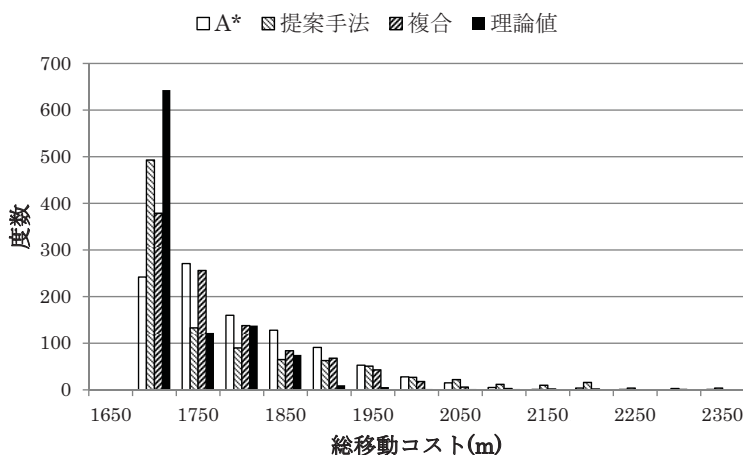


図5.16: 実験6-3の総移動コストのヒストグラム

また、3つの手法に関してコストの平均値に対する有意差検定の結果を表5.25に示す。

表5.25: 実験6-3におけるコストの平均の有意差検定結果

比較対象	t	P	有意差
A* : 提案手法	8.260	0.000	有
A* : 複合	6.691	0.000	有
提案手法 : 複合	3.426	0.000	有

5.6.3 考察

5.6.3.1 エージェント1体での実験結果考察

応用実験の結果より、提案手法の総移動コストがA*アルゴリズムの総移動コストより小さくなることがわかった。特に、A*アルゴリズムの総移動コストが理論値の総移動コストより大幅に大きくなる場合に、提案手法が効果的であることが多いという結果が得られた。これは、A*アルゴリズムが障害発生率を考慮していないせいで、A*アルゴリズムで計算された経路には、障害発生率の高い辺が存在してしまうことが多いのに対し、提案手法が障害発生率の高い辺を避けて経路を計算しており、提案手法で計算された経路上に、障害発生率の高い辺が存在しづらいことが理由だと考えられる。

また、A*アルゴリズムと提案手法の総移動コストの差がほとんどでない場合があるのは、グラフの障害発生率が全体的に低かったり、迂回経路が最短経路に比べて非常に長かったりして、A*アルゴリズムで計算した経路と提案手法で計算した経路が同じになってしまい、A*アルゴリズムと提案手法の差がなくなってしまうことが理由だと考えられる。

5.6.3.2 エージェント複数体での実験結果考察

応用実験の結果より、エージェントを複数体にした場合は、A*アルゴリズムと提案手法の差が小さくなることがわかった。これは、エージェントが複数

体いる場合に、A* アルゴリズムでは、目的地に近いエージェントが最短経路の障害の有無を確認するので、目的地から遠いエージェントは理論値とほぼ同じ経路で移動でき、A* アルゴリズムの総移動コストが理論値の総移動コストに近くなるが、提案手法では最短経路を最後まで通らない（最短経路の障害の有無を確認しない）ことがあり、エージェントを複数にしても、理論値の総移動コストに近くなることが少ないことが理由だと考えられる。

また、複合が A* アルゴリズムや提案手法よりも総移動コストが小さくなることがあるのは、提案手法の短所である「最短経路の障害の有無を確認しない」という特性を、A* アルゴリズムが消しているからだと考えられる。そして、複合が提案手法よりも総移動コストが大きくなることは、提案手法の短所である「最短経路の障害の有無を確認しない」という特性を A* アルゴリズムが消す利点よりも、A* アルゴリズムの短所である「障害に衝突しやすい」という特性が大きくなるからだと考えられる。

5.6.3.3 検定結果考察

検定の結果、多くの場合で有意差が有り、提案手法が優れていることがわかった。また、A* アルゴリズムの総移動コストが提案手法の総移動コストより小さくなる場合では、有意差は無かった。つまり、実際の地図データに提案手法を適用した場合でも効果があることがわかった。

第 6 章 まとめと今後の課題

6.1 まとめ

本論文では迂回経路を利用する手法を提案し、ほぼ全ての場合において、既存手法の総移動コストより、提案手法の総移動コストが小さくなるという結果を得ることができた。また、提案手法の総移動コストより、既存手法の総移動コストが小さくなる場合でも、有意差は無かった。このことから、本論文で提

案した手法は，あらかじめ障害情報を与えられていない状況において有効であることが分かった．したがって本手法は，地震災害を想定した道路ネットワークにおいて有効な手法であると考えられる．

6.2 今後の課題

今後の課題として，以下の2つが挙げられる

- 衝突制御

本論文では，複数体のエージェントで実験する際に，エージェント同士の衝突を考慮していない．しかし，エージェントを現実世界の車両等と想定した場合や，十分に広くない道路を移動すると想定した場合には，衝突制御が必要になると考えられる．

- 渋滞制御

本論文では，複数体のエージェントで実験する際に，エージェント同士での道路渋滞を考慮していない．しかし，エージェントを現実世界の車両等と想定した場合，渋滞制御が必要になると考えられる．

謝辞

本稿の一部は愛知大学研究助成（C-168）「災害救助エージェントシミュレーションにおける地図の定量化・分類と救助戦略の関連性」（2012年度）の成果である．

参考文献

- [1] Robocup rescue simulation, <http://roborescue.sourceforge.net/>.
- [2] P.E. Hart, N.J. Nilsson, and B. Raphael, “A formal basis for the heuristic determination of

- minimum cost paths”, Systems Science and Cybernetics, IEEE Transactions on, 4(2): pp. 100-107, 1968.
- [3] E.W.Dijkstra, “A note on two problems in connexion with graphs”, Numerische mathematic, 1(1): pp. 269-271, 1959.
- [4] A. Stentz, “Optimal and efficient path planning for partially-known environments”, In Robotics and Automation, 1994. Proceedings, 1994 IEEE International Conference on, pp. 3310-3317. IEEE, 1994.
- [5] S. Koenig and M. Likhachev, “A new principle for incremental heuristic search: Theoretical results”, In Proceedings of the International Conference on Automated Planning and Scheduling, pp. 410-413, 2006.
- [6] 国土地理院, 数値地図 25000.
- [7] あなたの街の地震マップ,
<http://www.city.nagoya.jp/kurashi/category/20-2-5-6-0-0-0-0-0-0-0.html>.
- [8] 永田靖, 吉田道弘, “統計的多重比較法の基礎”, サイエンティスト社, 1997